



SAVONIA

■ OPINNÄYTETYÖ – AMMATTIKORKEAKOULUTUTKINTO
TEKNIIKAN JA LIIKENTEEN ALA

KINECT-KUNTOUTUSSOVELLUS

Juti-hanke

TEKIJÄ: Terho Häkkinen

Koulutusala Tietotekniikka		
Koulutusohjelma Ohjelmistotekniikka		
Työn tekijä(t) Terho Häkkinen		
Työn nimi Kinect-kuntoutussovellus		
Päiväys	14.10.2013	Sivumäärä/Liitteet 23
Ohjaaja(t) Lehtori Martti Riikonen, ohjelmistosuunnittelija Mikko Pääkkönen		
Toimeksiantaja/Yhteistyökumppani(t) Savonia-AMK		
<p>Tiivistelmä</p> <p>Opinnäytetyön aiheena oli tehdä korjauksia vuonna 2012 toteutettuun Kinect-kuntoutussovellukseen. Työ, kuten alkuperäinen sovellus, tehtiin Juti-hanketta varten.</p> <p>Sovellukseen oli tehtävä korjauksia käytettävyyden parantamiseksi. Tehtäviin korjauksiin kuului ongelmat pelien liikkeentunnistuksessa, ongelmat sovelluksen valikoissa sekä ongelmat istuvan henkilön tunnistuksessa. Lisäksi sovelluksen peleihin tuli lisätä ohjeistukset käyttäjille.</p> <p>Opinnäytetyö toteutettiin Savonia-AMK:n TKI-tiloissa Kuopiossa. Työssä käytettiin C#-ohjelmointikieltä sekä Microsoftin XNA- ja Kinect-kehitystyökaluja.</p> <p>Opinnäytetyön tuloksena sovelluksesta saatiin valmiiksi korjattu versio, jonka jälkeen sovellus on ollut esillä ja yleisön vapaasti kokeiltavissa elokuussa 2013 Kuopiossa järjestetyssä Games for Health-tapahtumassa.</p>		
Avainsanat Kinect, C#, ohjelmointi, kuntoutussovellus, XNA		

Field of Study Technology, Communication and Transport		
Degree Programme Degree Programme in Information Technology		
Author(s) Terho Häkkinen		
Title of Thesis Kinect Rehabilitation Program		
Date	15 December 2013	Pages/Appendices 23
Supervisor(s) Mr Martti Riikkonen, Lecturer, Mr Mikko Pääkkönen, Software Designer		
Client Organisation /Partners Savonia UAS		
<p>Abstract</p> <p>The purpose of this thesis was to continue the work and fix problems in a rehabilitation application using Kinect for user motion tracking created in 2012. This thesis and the application were done for the Juti project.</p> <p>The application had problems in usability, which had to be fixed. The problems that needed to be fixed included problems in motion detection during the use of the software, problems in motion detection while navigating the application's menus and problems with detecting the user while he is sitting. In addition, the games in the applications needed instructions for the players.</p> <p>The thesis was implemented in the R&D department of Savonia University of Applied Sciences in Kuopio, Finland. Techniques used in the thesis included the C# -programming language along with Microsoft's XNA and Kinect Software Development Kits.</p> <p>After the problems had been solved and the thesis finished, the application was on show and free to try for the public during the Games for Health event held in Kuopio in August 2013.</p>		
<p>Keywords</p> <p>C#, programming, Kinect, XNA, rehabilitation application</p>		

SISÄLTÖ

TERMIT JA LYHENTEET.....	6
1 JOHDANTO.....	7
2 KINECT-SOVELLUS.....	8
2.1 SOVELLUKSESSA KÄYTETYT TEKNIIKAT.....	8
2.2 SOVELLUKSEN VAATIMUKSET.....	8
2.3 SOVELLUKSEN TARPEET LIIKKEENTUNNISTUKSESSA.....	9
2.4 SOVELLUKSEEN TOTEUTETUT PELIT.....	9
3 LIIKKEENTUNNISTUS.....	10
3.1 LIIKKEENTUNNISTUKSEEN SUUNNITELTUJA LAITTEITA.....	10
3.2 LIIKKEENTUNNISTUSTA HYÖDYNTÄVIÄ SOVELLUKSIA.....	11
4 MICROSOFT KINECT FOR WINDOWS.....	12
4.1 KINECTIN SENSORIT.....	12
4.1.1 KINECTIN INFRAPUNASENSORI.....	13
4.1.2 KINECTIN OPTINEN SENSORI.....	13
4.1.3 KINECTIN ÄÄNISENSORI.....	13
4.2 KÄYTTÄJÄN SEURAAMINEN KINECT -LAITTEELLA.....	14
5 TYÖN TOTEUTUS.....	15
5.1 KINECT-SOVELLUKSEN ONGELMAT.....	15
5.1.1 ONGELMIEN TUNNISTAMINEN.....	16
5.1.2 PELIEN OHJEISTUS.....	18
5.2 ONGELMIEN KORJAUS.....	18
5.2.1 ISTUVAN HENKILÖN TUNNISTAMINEN.....	18
5.2.2 PELIN OSOITTIMEN LIIKUTTAMINEN.....	19
5.2.3 HÄIRIÖT PELIN VALIKOISSA.....	20
6 YHTEENVETO.....	21
LÄHTEET	22

TERMIT JA LYHENTEET

SDK	Software development kit, eli ohjelmiston kehitystyökalut.
Redistributable	Uudelleenjaettava ohjelmisto.
Framework	Kokoelma kirjastoja ja luokkia ohjelmistoja varten.
CMOS-kenno	Digitaalikameroissa käytettävän valoherkän kennon tyyppi.
XNA	Microsoftin tarjoamat kehitystyökalut pelinkehitykseen Windows-, Xbox- ja Windows Phone-alustoille.
Unity	Unity Technologiesin kehittämä pelimoottori, joka toimii useimmilla PC- ja mobiilikäyttöjärjestelmillä.
DirectX	Grafiikka- ja audiotyökalut Windows- ja Xbox-alustoille.

1 JOHDANTO

Tarve työlle syntyi kesällä 2012 yhteistyössä fysioterapiaopiskelijoiden kanssa valmistuneen Juti-hanketta varten tehdyn Kinect-kuntoutussovelluksen ensimmäisestä toimivasta versiosta saadusta palautteesta. Sovelluksessa käytetään liikkeentunnistusta käyttäjän ja tämän liikkeiden seuraamiseen, jolloin käyttäjä voi ohjata sovelluksessa olevia pelejä sekä navigoida sovelluksen valikoissa ilman, että käyttäjän tarvitsisi käyttää lisälaitteita.

Työn tarkoituksena on parantaa kuntoutussovelluksen käytettävyyttä korjaamalla sovelluksen ensimmäiseen versioon jääneet ongelmat. Sovellusta kehitettiin Savonia-ammattikorkeakoulun TKI-tiloissa Kuopiossa. Sen suunnittelu tehtiin yhteistyössä Savonia-ammattikorkeakoulun terveysalan fysioterapiaopiskelijoiden kanssa. Sovellus toteutettiin projektina Juti-hanketta varten.

Sovellus oli esitteillä Kuopiossa huhtikuussa 2013 pidetyssä kansainvälisessä Coehre-kongressissa, jossa eri maista saapuneet opiskelijat pääsivät tutustumaan kuntoutuskäyttöön soveltuvia sovelluksia, sekä Games for Health -tapahtumassa Kuopiossa elokuussa 2013, jossa se oli vapaasti yleisön kokeiltavissa.

2 KINECT-SOVELLUS

Kinect-kuntoutussovellus kehitettiin vuoden 2013 touko-elokuu välisenä aikana. Sovelluksen kehitys tapahtui Savonia Ammattikorkeakoulun TKI-tiloissa, jotka sijaitsevat Kuopion Technopoliksella. Sovellus kehitettiin yhteistyönä ohjelmistotekniikan ja terveystieteiden opiskelijoiden kanssa Juti-hanketta varten.

Sovelluksen tarkoitus oli toimia käyttäjälle helppona ja mielekkäänä tapana omatoimiseen kuntouttavaan liikuntaan. Tämä saavutettiin tarjoamalla käyttäjälle erilaisia pelejä, jotka tukivat käyttäjän omatoimista liikuntaa hyödyntämällä liikkumista pelien ohjaamiseen. Peleissä ja valikoissa liikkumiseen ja niiden ohjaamiseen käytetään käyttäjän käsien liikkeistä ja sijainnista saatua tietoa, mutta sovelluksessa on mahdollisuus seurata myös käyttäjän muun vartalon liikkeitä.

2.1 SOVELLUKSESSA KÄYTETYT TEKNIIKAT

Sovellus on kehitetty käyttäen C#-ohjelmointikieltä. Sovelluksen tarvitsema Kinect-toiminnallisuus toteutettiin hyödyntämällä Microsoftin Kinect-SDK:ta. Kinect-toiminnallisuudella saatiin aikaiseksi sovelluksessa käytettävä liikkeentunnistus (Kinect).

Sovelluksessa käytettiin XNA-SDK:ta, jolla hoidettiin sovelluksen valikoiden sekä pelien sisäinen logiikka, sisällön kuten kuvien ja videoiden lisääminen sovellukseen sekä graafisten elementtien piirtäminen ruudulle. (XNA)

Sovelluksen kehitystyö tehtiin Microsoftin Visual Studio 2010 Professional-kehitysympäristössä. Sovelluksen kehitys tehtiin Savonian TKI-tiloissa Kuopion Technopoliksella.

2.2 SOVELLUKSEN VAATIMUKSET

Sovelluksen käyttämistä varten tarvitaan tietokone, jossa on vähintään Windows 7 -käyttöjärjestelmä ja johon on liitetty Microsoft Kinect for Windows -laite sekä näyttö. Lisäksi tietokoneeseen tulee olla asennettuna Microsoft Kinect Redistributable ja XNA Framework.

Lisäksi tietokoneessa on oltava vähintään 2 gigatavua keskusmuistia, 2.66 gigahertsin tuotteen prosessori, koneessa on oltava USB 2.0 -liitin ja tietokoneeseen on oltava asennettuna Microsoft Windows 7 tai Microsoft Windows 8 -käyttöjärjestelmä (Kinect system requirements).

2.3 SOVELLUKSEN TARPEET LIIKKEENTUNNISTUKSESSA

Koska sovellusta oli tarkoitus käyttää pääasiallisesti kuntoutustarpeisiin, liikkeentunnistuksella oli pystyttävä seuraamaan koko käyttäjää kolmiulotteisessa tilassa. Tästä syystä ohjaimelliset vaihtoehdot oli jätettävä pois, sillä niillä ei voitu seurata kuin käyttäjän käsiä. Koska osa kuntoutettavista saattoi olla huonokuntoisia, joille suuret liikkeet tuottaisivat vaikeuksia, oli sovelluksessa käytettävän liikkeentunnistuksen pystyttävä tunnistamaan myös pieniä ohjausliikkeitä.

2.4 SOVELLUKSEEN TOTEUTETUT PELIT

Sovellusta varten toteutettiin yhteensä kahdeksan eri peliä. Jokaisessa pelissä käyttäjä ohjaa ruudulla näkyviä erilaisia pelihahmoja käsillään. Osa peleistä on suunniteltu siten, että käyttäjä joutuu tekemään suuria liikkeitä tai liikkumaan paikaltaan pelatakseen, kun taas osassa peleistä käyttäjän on pystyttävä tekemään hillitympiä liikkeitä.

Esimerkkeinä toteutetuista peleistä on muun muassa Koppipeli, jossa pelaajan on onnistuttava nappaamaan puusta putoavia omenia käsillään ohjaamiin koreihin, ennen kuin ne putoavat maahan. Pelissä on aikaraja, jonka jälkeen peli laskee pisteet sen perusteella, kuinka monta omenaa pelaaja sai kiinni aikarajan puitteissa ja kuinka monta omenaa putosi maahan. Tässä pelissä pelaajan on tehtävä suuria liikkeitä, sillä pelaajan on liikutettava ohjaamiaan koreja koko peliruudun alueella.

Toinen esimerkki on Lentopeli, jossa pelaaja ohjaa helikopteria luolaston läpi. Pelaajan on pidettävä helikopteria ilmassa ja yritettävä olla osumatta luolaston seinämiin nostamalla ja laskemalla käsiään. Tässä pelissä käyttäjän ei tarvitse tehdä suuria liikkeitä, sillä pelin ohjaus perustuu siihen, kuinka korkealle pelaaja nostaa käsiään lantioltaan.

Kolmantena esimerkkinä on Siirtelypeli, jossa pelaajan pitää noukkia ja siirtää näytöllä olevia esineitä pelin antamien ohjeiden mukaisesti. Koppipelin tavoin tässä pelissä käytetään suuria liikkeitä, sillä pelaajan pitää noukkia ja siirtää pelin määrittämiä esineitä koko peliruudun alueella.

3 LIIKKEENTUNNISTUS

Liikkeentunnistusta käytetään mittaamaan ja seuraamaan käyttäjän kehon liikkeitä. Perusideana liikkeentunnistuksessa on tunnistaa kohteen sijainnin muutos suhteessa sen ympäristöön, tai ympäristön muutos kohteeseen nähden. Tähän tarvitaan aina sensoreita, joista saatua tietoa voidaan käyttää mittaamaan muutos tietyllä alueella. Liikettä voidaan tunnistaa hyödyntämällä mm. infrapunaa tai optiikkaa (videokamerat). Näitä on mahdollista yhdistää keskenään, jolloin on mahdollista saada tarkempia tuloksia.

Optisten sensorien käytön lisäksi on mahdollista käyttää myös lisävälineitä, jotka helpottavat kohteen sijainnin tunnistamista sensoriin nähden. Nämä lisävälineet on yleensä kiinnitettävä käyttäjään, tai käyttäjän on itse pidettävä niitä käsissään.

3.1 LIIKKEENTUNNISTUKSEEN SUUNNITELTUJA LAITTEITA

Perinteisin esimerkki liikkeentunnistuksesta ovat eri kauppojen automaattisesti aukeavat liukuovet. Nämä toimivat yleensä eräänlaisella tutkalla, joka lähettää mikroaaltoradioenergiapurkauksia halutulle alueelle ja odottaa lähettämänsä energian heijastumista takaisin. Kun henkilö kävelee tälle alueelle, tutkan vastaanottama energiamäärä tai aika, joka kuluu energian palaamiseen tutkalle muuttuu, jolloin tutka lähettää oville avautumiskäskyn (How do motion sensing lights and burglar alarms work? 2000).

Toinen esimerkki liikkeentunnistuksesta on pelikonsoleita varten suunnitellut ohjaimet, kuten Nintendon Wiimote ja Sonyn Playstation Move-ohjaimet sekä Microsoftin Xbox 360 -konsolille alunperin suunniteltu Kinect-sensori. Näistä kahdessa, Wiimote- ja Playstation Move -ohjaimissa, käyttäjän on pideltävä käsissään ohjausta ja liikkeentunnistusta varten käytettävää ohjainta. Xbox 360 -konsolin liikkeentunnistusta varten tarvitaan ainoastaan sensorina toimiva Kinect.

Näiden lisäksi on PC- ja MAC- tietokoneille kehitetty Leap Motion -ohjain, jota voidaan käyttää tietokoneen ohjaamiseen. Leap Motion -ohjain seuraa käyttäjän käsien liikkeitä sen havaitsemalla alueella, eikä sitä olla suunniteltu koko käyttäjän kehon seuraamiseen. Koska liikkeiden seuraaminen on rajattu käyttäjän käsiin, Leap Motion pystyy seuraamaan tarkasti käyttäjän liikkeitä (Leap Motion).

3.2 LIIKKEENTUNNISTUSTA HYÖDYNTÄVIÄ SOVELLUKSIA

Liikkeentunnistusta voi käyttää sovelluksissa moneen tarkoitukseen. Pelien lisäksi eri esimerkkeinä voitaisiin nostaa Philipp Robbelin kehittämä KinectBot, jossa hän yhdisti Kinectin iRobot Create -laitteeseen, jolloin iRobot pystyi kartoittamaan tilaa 3D:nä ja ymmärtämään ihmisten sille esittämiä eleitä. Lisäksi Kinectille on kehitetty erilaisia sovelluksia, joiden avulla käyttäjät voivat piirtää kolmiulotteisia kuvia ja manipuloida niitä sekä sovelluksia, joilla on mahdollista käsitellä erilaisia virtuaalisia esineitä, kuten nukkeja. (Wortham 2010).

Liikkeentunnistusta voidaan myös hyödyntää käyttöliittymien kanssa. Tästä voitaisiin mainita esimerkkinä Googlen Chrome-selainta varten tehty DepthJS-liitännäinen, joka mahdollistaa selaimen käyttämisen hyödyntämällä Kinectiä käyttäjän tekemien eleiden tunnistamiseen (DepthJS).

Hyvänä kaupallisena esimerkkinä voitaisiin käyttää hetken aikaa Moskovassa sijaitsevassa TopShop -vaateliikkeessä ollutta vaateensovitussovellusta, jossa käyttäjä pystyi kokeilemaan päälleen eri vaatteita virtuaalisesti käyttäen Kinect-laitetta ja tähän liitettyä näyttöä (Citizen 2011).

4 MICROSOFT KINECT FOR WINDOWS

Alunperin pelikäyttöön tarkoitettu Kinect julkaistiin vuonna 2010 Microsoftin Xbox -konsolille (Whitworth 2010). Vuonna 2011 Microsoft julkaisi Kinectistä kehitystyökalut Windows 7 -käyttöjärjestelmälle, jonka myötä Kinect on siirtynyt pelkästä peliohjaimesta laajempaan käyttöön (Academics, Enthusiasts to Get Kinect SDK).

Microsoft on myös kehittämässä tällä hetkellä seuraavaa versiota Kinect for Windowsista, joiden kehittäjille suunnatut ennakkoesittelypaketit on jo toimitettu hyväksytyille osanottajille (Thousands of developers are participating in Kinect for Windows v2 Developer Preview-starting today).

4.1 KINECTIN SENSORIT

Kinectin toiminta perustuu laitteessa oleviin sensoreihin, joihin kuuluu videokamera (optinen sensor), laserpisteprojektori ja CMOS-kenno (infrapunasensori) sekä neljä mikrofonia (äänisensori). Alla olevassa kuvassa (KUVA 1) on merkittynä sensorien sijainti Kinectissä. Näitä voidaan käyttää yhdessä, jotta saataisiin tarkka kuva siitä, missä käyttäjä on itse laitteeseen nähden (Kinect for Windows Sensor Components and Specifications).



KUVA 1. Kinect-sensori. Kuvassa merkittynä Kinectin sisältämät sensorit: 1. Laserpisteprojektori 2. videokamera 3. CMOS-kenno 4. äänisensori

4.1.1 KINECTIN INFRAPUNASENSORI

Kinectin infrapunasensorissa on kaksi osaa: laserpisteprojektori ja CMOS-kenno. Sensorin toiminta perustuu näiden kahden osan yhteistyöhön. Laserpisteprojektori lähettää pistemäisiä valoja sensorin edessä olevalle alueelle, jotka heijastuvat takaisin CMOS-kennolle, jolloin sensorille välittyy kuva laserpisteprojektorin lähettämistä valopisteistä. Tästä kuvasta sensori pystyy laskemaan eri kohteiden, kuten esineiden tai pelaajien, päälle osuneiden valopisteiden etäisyyden sensoriin nähden ja muodostaa näistä ns. syvyyskartan (Kinect for Windows Sensor Components and Specifications).

Syvyyskartta on hyvin samanlainen normaalin kuvatiedoston kanssa, mutta sen sijaan, että kuvan pikselissä olisi tieto sen väristä, on kuvakartassa pikselin kohdalla tieto sen etäisyydestä. Syvyyskartta on saatavilla kolmella eri resoluutiolla: 320x240, 640x480 ja 800x600 pikseliä (Depth Stream).

4.1.2 KINECTIN OPTINEN SENSORI

Kinect sisältää VGA-kameran, jota käytetään käyttäjän kasvojen tunnistamiseen ja videopuheluihin. Kameralta pystytään ottamaan videokuvaa 1280x960 pikselin resoluutiolla ja se pystyy kuvaamaan 30 kuvaa sekunnissa. Halutessaan Kinectiä hyödyntäviin sovelluksiin voidaan lisätä suoraan Kinectin VGA-kameran kuvaama alue, jolloin pelaajat voivat nähdä itsensä samalla, kun he käyttävät sovelluksia (Kinect for Windows Sensor Components and Specifications)).

4.1.3 KINECTIN ÄÄNISENSORI

Kinect sisältää neljä mikrofonia, joita voidaan käyttää äänikomentojen antamiseen. Äänisensorilla on myös mahdollista seurata, mistä suunnasta ääni kuuluu sensoriin nähden ja se pystyy suodattamaan puheesta ylimääräisen kohinan ja taustääänet (Kinect for Windows Sensor Components and Specifications)).

4.2 KÄYTTÄJÄN SEURAAMINEN KINECT -LAITTEELLA

Kinectissä on kaksi eri seurantatilaa, joilla käyttäjää voidaan seurata joko istuvasta tai seisovasta asennosta, sekä kaksi eri etäisyysasetusta, joilla saadaan aikaiseksi parempi tarkkuus käyttäjän liikkeiden seuraamiseen sekä lähietäisyydeltä että kauempaa. Kinect ei itse vaihda näiden asetusten välillä, joten vaihtaminen käyttäjän seuraamisesta lähietäisyydeltä normaaliin etäisyyteen on hoidettava ohjelmiston puolelta samoin kuin seurantatilan vaihtaminen (Tracking Modes (Seated and Default)).

Kinect muodostaa käyttäjästä keräämästään datasta luurankomalliin sijoitettavat *nivelet*, jotka toimivat samaan tapaan kuin käyttäjän omat nivelet kuten olka- tai kyynärpää. Jokainen nivel sisältää tiedon nivelen sijainnista, kulmasta sekä etäisyydestä Kinect-laitteeseen. Käyttäjistä muodostettava luurankomalli kattaa joko koko kehon tai kehon yläosan, johon kuuluu pää, hartiat ja kädet, riippuen käytetäänkö käyttäjän seuraamiseen normaalia- vai istumatilaa (Tracking Skeletons in Near Depth Range).

Kinect for Windows -laitteella on mahdollista seurata käyttäjää kahdella etäisyysasetuksella. Lähietäisyysasetus mahdollistaa käyttäjän seuraamisen viidenkymmenen sentin ja kolmen metrin välillä, normaali etäisyysasetus mahdollistaa käyttäjän seuraamisen kahdeksankymmenen sentin ja neljän

metrin välillä. Näiden rajojen ulkopuolelle siirtyminen estää Kinect-laitetta muodostamasta käyttäjästä luurankomallia, jolloin käyttäjän seuraaminen ei enää ole mahdollista. Käyttäjän seuraaminen jatkuu, kun käyttäjä siirtyy takaisin etäisyydelle josta Kinect pystyy muodostamaan käyttäjästä luurankomallin (Tracking Skeletons in Near Depth Range).

5 TYÖN TOTEUTUS

Opinnäytetyön tarkoituksena oli korjata Kinect-kuntoutussovelluksessa olleita ongelmia sekä pari pienempää muutosta, joita sovellukseen toivottiin testauksen jälkeen. Työn tavoitteena on kehittää potilaille suunnattu kuntoutussovellus fysioterapiseen kuntoutukseen.

Työn aikana tehty ongelmien tunnistaminen suoritettiin toistamalla ongelmakohdat sovellusta käyttämällä. Tähän tarkoitukseen käytettiin tietokonetta, johon sovellus ja sen suorittamiseen tarvittavat ajurit oli asennettu, sekä Kinect for Windows -laitetta, jota käytettiin liikkeentunnistukseen.

Ongelmien korjaukseen käytettiin tietokonetta, johon oli asennettu Kinect for Windows -laite, sen käyttämiseen tarkoitetut ajurit, Kinect -kehitystyökalut, XNA 4.0 kehitystyökalut sekä Microsoft Visual Studio 2010 -kehitysympäristö, jolla varsinainen työ tehtiin.

5.1 KINECT-SOVELLUKSEN ONGELMAT

Kinect-sovelluksen ensimmäisen toimivan version testauksen jälkeen huomattiin, että sovelluksen ongelmiin kuuluivat mm. häiriöt pelin valikoissa liikkeessa, pelien osoittimien liikuttamisessa ilmaantuneet virheet ja istuvan pelaajan tunnistus. Testauksen jälkeen esitettiin myös toivomus, että sovellukseen lisättäisiin eri pelien ohjeistus.

5.1.1 ONGELMIEN TUNNISTAMINEN

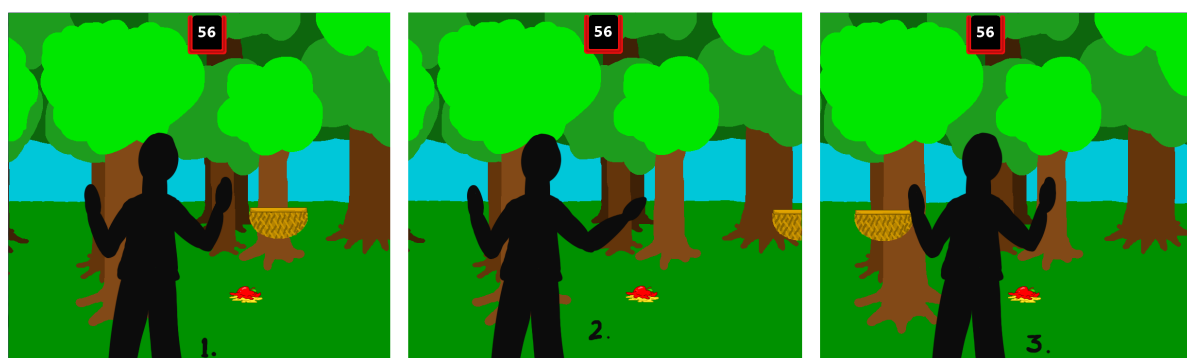
Aloitin työn kokeilemalla sovellusta löytääkseni ennen työn aloittamista todetut ongelmat, jotta saisin paremman käsityksen siitä mitä minun tulisi tehdä.

Ensimmäinen tunnistettu ongelma oli häiriöt pelin valikoissa liikkuesssa. Käyttäjän oli vaikeaa käynnistää pelejä ja liikkua valikoissa, sillä Kinect-laitteen saama tieto käyttäjästä ei ollut täysin tarkkaa, vaan sijaintitiedot käyttäjän käsien sijainnista vaihtelivat jokaisen mittauksen välillä (KUVA 2).



KUVA 2. Valikkonäkymä. Ruudulla näkyvä neliö on osoitin, jolla käyttäjä liikkuu valikoissa. Neliön sisällä olevat punaiset pisteet esittävät kuinka paljon osoitin liikkuu, vaikka käyttäjä pysyisi paikallaan.

Seuraavaksi tunnistin ongelman pelien osoittimien liikuttamisessa. Peleissä, joissa liikuttiin paljon, kävi usein niin, ettei pelaajan ohjaamien osoittimien sijainti ollut täysin yhdenmukaista, varsinkaan silloin kun pelaaja kurotti yli ns. ”pelialueen”. Alla olevassa kuvassa havainnoidaan kyseistä ongelmaa tarkemmin (KUVA 3). Tästä ongelmasta johtuen pelaaja joutui korjaamaan osoittimien sijaintia pelatessaan, jolloin pelaaminen hankaloitui ja tuntui turhautavalta.



KUVA 3. Ongelma pelien osoittimien liikuttamisessa. Kohdassa 1 ollaan alkupisteessä, jonka jälkeen pelaaja siirtää kätensä sivuun kohdan 2 osoittamalla tavalla, vieden osoittimen pelialueen reunalle. Siirtäessään kätensä takaisin alkupisteeseen kohdan 3 mukaisesti, osoitin oli siirtynyt selkeästi eri kohtaan kuin missä se oli ollut kohdassa 1.

Lopuksi tunnistin ongelman istuvan pelaajan tunnistamisessa. Pelaajan oli hankalaa päästä pelaamaan, jos tämä istui ennen kuin Kinect oli tätä tunnistanut. Useimmiten tunnistus tapahtui vasta, kun pelaaja oli seisomassa tai oli nousemassa tuolilta seisomaan. Tämä oli hankalaa, varsinkin jos pelaaja ei pystynyt seisomaan esimerkiksi liian heikon kunnon tai vakavan vamman vuoksi (esimerkiksi pyörätuolipotilaat).

5.1.2 PELIEN OHJEISTUS

Pelien ohjeistus oli yksi lisätoiveista, joka esitettiin ensimmäisen valmiin version testauksen jälkeen. Vaikka pelit eivät olleet monimutkaisia, oli pelaajille kerrottava eri pelien säännöistä ja pelaajien tavoitteista.

Ohjeistukset toteutettiin peliin omana ruutunaan, joka näytettiin pelaajalle tämän käynnistettyä jonkun peleistä. Ohjeet tulivat pelaajan näkyville tekstinä, josta pelaaja näki tehtävän ja säännöt. Ohjeet luettuaan pelaaja pystyi siirtymään itse peliin siirtämällä ruudulla olevan kohdistimen ruudulla näkyvän kehoitelaatikon päälle.

5.2 ONGELMIEN KORJAUS

Kun sovelluksessa olevat ongelmat oli tunnistettu, voitiin aloittaa mahdollisten ratkaisujen etsiminen. Ongelmien ratkaisu toteutettiin ongelma kerrallaan, kokeilemalla muutoksia lähdekoodiin ja kyseisten muutosten vaikutus todettiin käynnistämällä sovellus ja kokeilemalla sovellusta muutosten toteutuksen jälkeen.

5.2.1 ISTUVAN HENKILÖN TUNNISTAMINEN

Ratkaisun hakeminen aloitettiin tutkimalla sovellukseen lähdekoodia, eritoten siellä olevaa Kinect-laitteeseen liittyvää koodia. Lähdekoodista huomattiin, ettei sovellukseen oltu lisätty mahdollisuutta vaikuttaa, mitä seurantatilaa Kinect käyttää, vaan se oltiin asetettu olemaan aina tilassa, jossa seurattaisiin seisaallaan olevaa ihmistä.

Lähdekoodia muuttamalla Kinect saatiin tilaan, jossa se seuraa istuvaa ihmistä, mutta ongelmaksi muuttui koko käyttäjän kehon seuraamisen puuttuminen. Ratkaisuksi sovellukseen lisättiin mahdollisuus vaihtaa tätä tilaa sovelluksen asetuksista, johon käyttäjä pääsee käsiksi käyttäen sovelluksen omaa graafista valikkoa. Lisäksi lähdekoodiin lisättiin muutos sen käynnistyksen yhteyteen, joka asettaa istuvan hahmon tunnistuksen tilaksi, jos sovellus käynnistetään ensimmäistä kertaa.

5.2.2 PELIN OSOITTIMEN LIKUTTAMINEN

Ongelmaa tunnistettaessa huomattiin, että ongelma toistui kaikissa sovelluksen peleissä. Tästä pääteltiin, ettei ongelma ollut yksittäisen pelin lähdekoodissa, vaan ongelma johtui mahdollisesti tavasta, jolla pelaajan liikkeet muutettiin pelin osoittimien liikkeiksi. Siispä ongelman ratkaisua aloitettiin hakemaan sovelluksen lähdekoodin osasta, jossa Kinectiltä saatu pelaajan sijaintidata muutettiin sovellukseen sopivaan muotoon.

Lähdekoodia tarkemmin tutkiessa huomattiin tapa, jolla osoittimen liikkuminen ruudulla näkyvän pelialueen ulkopuolelle estettiin. Kun osoitin oli pelialueen reunalla, sitä ei enää liikutettu suuntaan, josta se olisi ylittänyt pelialueen. Tämän jälkeen, pelaajan liikuttaessa osoitinta pois pelialueen reunalta, osoittimen liikkuminen ruudulla ei ollut yhdenmukaista pelaajan aiemman sijaintitiedon kanssa.

Lähdekoodia muokattiin aluksi poistamalla osoittimen liikuttamisen rajoitus, mutta tämä aiheutti lisää ongelmia pelien ohjattavuudessa, joten tehdyt muutokset peruttiin. Kokeilemalla osoittimen rajoituksien muuttamista osoittimen liikkumista saatiin korjattua toivotunlaiseksi. Tämän jälkeen sovelluksen eri pelejä testattiin tehdyillä muutoksilla, jonka jälkeen lähdekoodia muokattiin, jotta osoittimen liikuttaminen saataisiin toimimaan

toivotulla tavalla. Parhaaksi ratkaisuksi todettiin osoittimen rajoitusten muokkaaminen koko osoittimen liikuttamisen uudelleenohjelmoimisen sijasta.

5.2.3 HÄIRIÖT PELIN VALIKOISSA

Koska ongelma kohdistui pelin valikkoihin, lähdettiin ratkaisua hakemaan valikoiden lähdekoodista. Lähdekoodista löydettiin tapa, jolla valikoiden kohteiden, kuten esimerkiksi eri pelien, valinta oltiin toteutettu. Sovellus odotti tilannetta, jolloin pelaaja oli ohjannut ruudulla näkyvän osoittimen valikkoelementin, kuten pelivaihtoehdon, päälle ja pitänyt sitä paikallaan. Tietyn ajan kuluttua osoittimen oltua paikallaan sovellus alkaa laskemaan aikaa, jonka jälkeen sovellus olettaa pelaajan tehneen valintansa. Prosessi kuitenkin voitiin keskeyttää liikuttamalla osoitinta ruudulla.

Valintaprosessi olisi muuten toimiva, mutta valintaa oli hankala aloittaa ja suorittaa loppuun ohjauksessa tapahtuvien pienten häiriöiden vuoksi. Ratkaisuksi valikkoelementin valinnan lähdekoodiin lisättiin rajoitus, joka mahdollisti osoittimen liikuttamisen valintaprosessin aikana. Tätä rajoitusta muokattiin, kunnes löydettiin toimiva häiriöraja. Vaihtoehtoinen ratkaisu olisi ollut Kinectiltä saatavan datan tarkempaa suodattamista, mutta kyseinen ratkaisu olisi tehnyt osoittimen liikuttamisesta hitaampaa joten sitä ei käytetty.

6 YHTEENVETO

Työn tavoitteena oli korjata kesällä 2012 harjoittelijoiden toteuttamassa Kinect-kuntoutussovelluksessa olleita vikoja, sekä lisätä sovelluksen peleille ohjeistukset sovellukseen. Korjausten tekemisellä ja ohjeiden lisäämisellä oli tarkoitus parantaa sovelluksen käytettävyyttä ja tehdä sovelluksesta näin parempi potilaskäyttöön.

Työn tuloksena saatiin korjattua ennen työn aloittamista löytyneet viat, sekä sovellukseen saatiin lisättyä ohjeistukset peleihin. Työn valmistumisen jälkeen sovellus oli esitteillä ja yleisön vapaasti kokeiltavissa elokuussa 2013 Kuopiossa järjestetyssä Games for Health-tapahtumassa, jossa sovelluksen vastaanotto oli positiivista.

Mahdollisuus päästä työskentelemään liikkeentunnistuksen ja pelinkehityksen parissa tarjosi hyvin omanlaisensa haasteet, jotka tekivät työstä mielenkiintoisen. Suuri osa työn parissa kulutetusta ajasta kului sovelluksen lähdekoodin tutkimiseen ja kehitystyökaluihin tutustuen. Vikojen testaaminen, korjaaminen ja korjauksien testaaminen sujui vauhdikkaasti, jonka myötä työkin valmistui nopealla aikataululla. Olen tyytyväinen työni lopputulokseen, etenkin valmistumisen jälkeen saadun palautteen perusteella.

Sovelluksessa on edelleen joitain korjaamattomia ongelmakohtia, joita ei työni puolesta korjattu. Sovelluksen jatkokehitykseen kuuluukin siis ehdottomasti ongelmien korjaus, mutta sovellusta voisi jatkokehittää myös lisäämällä sovellukseen uusia pelejä. Jatkokehityksen kannalta olisi myös hyvä harkita tukea 3D-grafiikalle, joko lisäten ominaisuuden XNA:lla tai vaihtamalla projektissa käytetty XNA-framework Unity-pelimoottoriin.

LÄHTEET

Academics, Enthusiasts to Get Kinect SDK. [verkkosivu].[viitattu 20.12.2013].

Saatavissa <http://research.microsoft.com/en-us/news/features/kinectforwindowssdk-022111.aspx>

CITIZEN, Jessica 2011-05-10. Kinect used as Topshop dressing room. Player Attack. [viitattu 15.7.2013]. Saatavissa:

<http://www.playerattack.com/news/2011/05/10/kinect-used-as-topshop-dressing-room/>

DepthJS. [verkkosivu].[viitattu 18.12.2013]. Saatavissa

<https://github.com/doug/depthjs>

Depth Stream. [verkkosivu].[viitattu 20.12.2013]. Saatavissa

<http://msdn.microsoft.com/en-us/library/jj131028.aspx>

HowStuffWorks, 2000-01-04. How do motion sensing lights and burglar alarms work? [verkkosivu].[viitattu 20.12.2013]. Saatavissa

<http://home.howstuffworks.com/home-improvement/household-safety/security/question238.htm>

Kinect. [verkkosivu].[viitattu 19.12.2013]. Saatavissa

<http://www.microsoft.com/en-us/kinectforwindows/>

Kinect for Windows Sensor Components and Specifications. [verkkosivu].

[viitattu 20.12.2013]. Saatavissa <http://msdn.microsoft.com/en-us/library/jj131033.aspx>

Kinect system requirements. [verkkosivu].[viitattu 20.12.2013]. Saatavissa

<http://msdn.microsoft.com/en-us/library/hh855359.aspx>

Leap Motion. [verkkosivu].[viitattu 18.12.2013]. Saatavissa
<https://leapmotion.com>

Thousands of developers are participating in Kinect for Windows v2 Developer Preview—starting today. [verkkosivu].[viitattu 20.12.2013]. Saatavissa
<http://blogs.msdn.com/b/kinectforwindows/archive/2013/11/22/thousands-of-developers-are-taking-part-in-kinect-for-windows-v2-developer-preview.aspx>

Tracking Modes (Seated and Default). [verkkosivu].[viitattu 20.12.2013].
 Saatavissa <http://msdn.microsoft.com/en-us/library/hh973077.aspx>

Tracking Skeletons in Near Depth Range. [verkkosivu].[viitattu 20.12.2013].
 Saatavissa <http://msdn.microsoft.com/en-us/library/hh855353.aspx>

Tracking Users with Kinect Skeletal Tracking. [verkkosivu].[viitattu
 20.12.2013]. Saatavissa <http://msdn.microsoft.com/en-us/library/jj131025.aspx>

WHITWORTH, Dan 2010-08-17. Kinect gets UK release date. BBC Newsbeat.
 [viitattu 15.7.2013]. Saatavissa: <http://www.bbc.co.uk/newsbeat/10996389>

WORTHAM, Jenna 2010-11-21. With Kinect Controller, Hackers Take
 Liberties. New York Times. [viitattu 15.7.2013]. Saatavissa:
http://www.nytimes.com/2010/11/22/technology/22hack.html?_r=2&

XNA. [verkkosivu].[viitattu 19.12.2013]. Saatavissa
<http://msdn.microsoft.com/en-us/aa937791.aspx>